

JAVA.LANG.INTEGER CLASS

http://www.tutorialspoint.com/java/lang/java_lang_integer.htm

Copyright © tutorialspoint.com

Introduction

The **java.lang.Integer** class wraps a value of the primitive type `int` in an object. An object of type `Integer` contains a single field whose type is `int`.

Class declaration

Following is the declaration for **java.lang.Integer** class:

```
public final class Integer
    extends Number
    implements Comparable<Integer>
```

Field

Following are the fields for **java.lang.Integer** class:

- **static int MAX_VALUE** -- This is a constant holding the maximum value an `int` can have, $2^{31}-1$.
- **static int MIN_VALUE** -- This is a constant holding the minimum value an `int` can have, -2^{31} .
- **static int SIZE** -- This is the number of bits used to represent an `int` value in two's complement binary form.
- **static Class<Integer> TYPE** -- This is the class instance representing the primitive type `int`.

Class constructors

S.N.	Constructor & Description
1	Integer(int value) This constructs a newly allocated <code>Integer</code> object that represents the specified <code>int</code> value.
2	Integer(String s) This constructs a newly allocated <code>Integer</code> object that represents the <code>int</code> value indicated by the <code>String</code> parameter.

Class methods

S.N.	Method & Description
1	<u>static int bitCount(int i)</u> This method returns the number of one-bits in the two's complement binary representation of the specified <code>int</code> value.
2	<u>byte byteValue()</u> This method returns the value of this <code>Integer</code> as a byte.

3	<u>int compareTo(Integer anotherInteger)</u> This method compares two Integer objects numerically.
4	<u>static Integer decode(String nm)</u> This method decodes a String into an Integer.
5	<u>double doubleValue()</u> This method returns the value of this Integer as a double.
6	<u>boolean equals(Object obj)</u> This method compares this object to the specified object.
7	<u>float floatValue()</u> This method returns the value of this Integer as a float.
8	<u>static Integer getInteger(String nm)</u> This method determines the integer value of the system property with the specified name.
9	<u>static Integer getInteger(String nm, int val)</u> This method determines the integer value of the system property with the specified name.
10	<u>static Integer getInteger(String nm, Integer val)</u> This method returns the integer value of the system property with the specified name.
11	<u>int hashCode()</u> This method returns a hash code for this Integer.
12	<u>static int highestOneBit(int i)</u> This method returns an int value with at most a single one-bit, in the position of the highest-order ("leftmost") one-bit in the specified int value.
13	<u>int intValue()</u> This method returns the value of this Integer as an int.
14	<u>long longValue()</u> This method returns the value of this Integer as a long.
15	<u>static int lowestOneBit(int i)</u> This method returns an int value with at most a single one-bit, in the position of the lowest-order ("rightmost") one-bit in the specified int value.
16	<u>static int numberOfLeadingZeros(int i)</u> This method returns the number of zero bits preceding the highest-order ("leftmost") one-bit in the two's complement binary representation of the specified int value.
17	<u>static int numberOfTrailingZeros(int i)</u> This method returns the number of zero bits following the lowest-order ("rightmost") one-bit in the two's complement binary representation of the specified int value.
18	<u>static int parseInt(String s)</u> This method parses the string argument as a signed decimal integer.
19	<u>static int parseInt(String s, int radix)</u> This method parses the string argument as a signed integer in the radix specified by the second argument.
20	<u>static int reverse(int i)</u> This method returns the value obtained by reversing the order of the bits in the two's complement binary

	representation of the specified int value.
21	<u>static int reverseBytes(int i)</u> This method returns the value obtained by reversing the order of the bytes in the two's complement representation of the specified int value.
22	<u>static int rotateLeft(int i, int distance)</u> This method returns the value obtained by rotating the two's complement binary representation of the specified int value left by the specified number of bits.
23	<u>static int rotateRight(int i, int distance)</u> This method returns the value obtained by rotating the two's complement binary representation of the specified int value right by the specified number of bits.
24	<u>short shortValue()</u> This method returns the value of this Integer as a short.
25	<u>static int signum(int i)</u> This method returns the signum function of the specified int value.
26	<u>static String toBinaryString(int i)</u> This method returns a string representation of the integer argument as an unsigned integer in base 2.
27	<u>static String toHexString(int i)</u> This method returns a string representation of the integer argument as an unsigned integer in base 16.
28	<u>static String toOctalString(int i)</u> This method returns a string representation of the integer argument as an unsigned integer in base 8.
29	<u>String toString()</u> This method returns a String object representing this Integer's value.
30	<u>static String toString(int i)</u> This method returns a String object representing the specified integer.
31	<u>static String toString(int i, int radix)</u> This method returns a string representation of the first argument in the radix specified by the second argument.
32	<u>static Integer valueOf(int i)</u> This method returns a Integer instance representing the specified int value.
33	<u>static Integer valueOf(String s)</u> This method returns an Integer object holding the value of the specified String.
34	<u>static Integer valueOf(String s, int radix)</u> This method returns an Integer object holding the value extracted from the specified String when parsed with the radix given by the second argument.

Methods inherited

This class inherits methods from the following classes:

- java.lang.Object

